Proceedings of the Joint Austrian Computer Vision and Robotics Workshop 2020 DOI: 10.3217/978-3-85125-752-6-07

Automatic Ontology-based Plan Generation for an Industrial Robotics System

Timon Hoebert, Wilfried Lepuschitz, Munir Merdan
Practical Robotics Institute Austria
{hoebert,lepuschitz,merdan}@pria.at

Abstract. *Programming and re-configuration of robots are associated with high costs, especially for small- and medium-sized enterprises. We present an ontology-driven solution that can automate the configuration as well as the generation of process plans and schedules thereby significantly lowering the efforts in the case of changes. The presented approach is demonstrated in a laboratory environment with an industrial pilot test case.*

1. Introduction

Robotics technology, which can prove high efficiency, precision, and repeatability, is regarded as a viable solution to cope with the increasing number of individualized products. However, robot systems still often do not meet the demands of small- and medium-sized enterprises (SMEs) [8]. Especially, since the programming of industrial robots is complex and time-consuming. To be able to dynamically adapt to new products, robotic systems need to work autonomously. Autonomous systems, in this context, means that robots systems can perform high-level task specifications without explicitly being programmed [2]. To reach specific goals, such systems should be able to receive goals and automatically sequence plans and execute them considering their current state. In our previous work, we presented the control architecture for industrial robots, which can generate actions based on an product model by linking product model, manufacturing process, and production environment in an ontology [7]. In this paper, we focus on the automated plan generation from the ontology and present an approach for flexibly coupling of the decision-making mechanism and ontology.
In section 2, we will detail the architecture and implementation. Finally, Section 3 concludes the paper with a summary and an outlook on further research issues.

2. Architecture

The industrial robot control layer responsible for the management of the robotics systems consists of a World-Model and a Decision-Making component. The decision-making mechanism (Planner) acts as a link between the semantic model of the production environment and the available robot system capabilities. The World Model contains the semantic representation of the relevant objects in the robotics system including their properties and relations. The Planning Domain Definition Language (PDDL) is used for decision-making and the world model is conceptually defined using the Web Ontology Language (OWL) standard. In this context, we transform robotics domain knowledge represented in OWL to PDDL as a targeted mechanism for planning. Multiple applied robotic systems use PDDL for task planning and a lot of work has been done in combining ontologies and AI planning base [5, 1]. Especially ROSPlan [4], a ROS implementation, is a commonly used implementation for this purpose. Based on ROSPlan, OWL-ROSPlan [3] extends this approach using a specialized OWL-Ontology as knowledge-base instead of traditional databases. The disadvantage of these approaches is the implementation effort for application. Even OWL-ROSPlan requires a predefined data format of the ontology. Our work extends this research by automating the translation of the input required by PDDL from the ontology as well as from the PDDL back to the ontology without any predefined ontology formats.

2.1. OWL-PDDL Mapping scheme

The basic building blocks of OWL, are triples consisting of subject, predicate, and object. The basic building blocks of PDDL are actions and PDDL-predicates. To avoid confusion of the two different

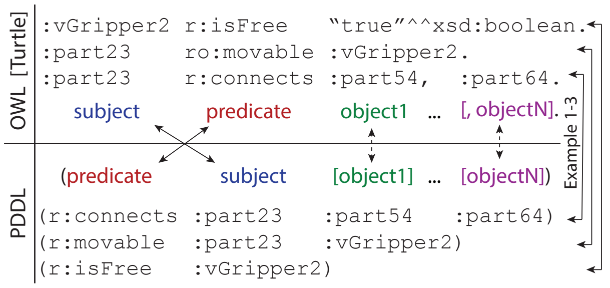

Figure 1. Mapping OWL triples with PDDL predicates. Also, three examples of different parameter length are shown.

types of predicates, the latter ones are only referred to as PDDL-Predicates. The general idea of this approach is the equalization of both building blocks, relating triples with PDDL-predicates. Using a similar approach like WebPDDL [6], OWL-IRIs are used as PDDL-predicate names to identify the data distinctly.

OWL-predicates relate subjects and objects, as verbs do in sentences, but PDDL-predicates are only binary statements relating to multiple object parameters. In practice, PDDL-predicates usually only have one or two object parameters, which can be seen as subject and object. The complete mapping scheme is illustrated with three examples in Figure 1. PDDL-predicates with only one parameter are mapped to boolean-valued objects triples. In practice, PDDL-predicates with more than two parameters are rare because of their complexity (only 4 percent of all predicates from all IPC (1998-2018) domains. But, even these PDDL-predicates can be simplified to multiple PDDL-predicates with two parameters.

2.2. Semantic PDDL Generation

The system automatically generates the PDDL3-problem for the planner based on the information in the ontology and PDDL-domain. This enables easy and extensible programming of the system. The user only has to specify the PDDL-domain with IRIs as PDDL-predicate names and add the goal as triples into a separate part (separate graph) of the ontology database. The system automatically queries all triples of NamedIndividuals regarding this predicates, maps them to PDDL-predicates as mentioned earlier and adds them to the init section in the PDDL-problem. These queries are executed in parallel, and the particular subjects are recorded. After querying the triples, the OWL-types of the recorded subjects are searched in the ontology and written into the PDDL-problem. Since each NamedIndividual can have multiple parent-classes, but not all are relevant for planning, only the ones which are specified in the PDDL-domain are used.

3. Conclusion

The proposed knowledge-driven approach simplifies the programming efforts of the industrial robot. The code for the industrial implementation is generated automatically based on the defined rules, states and actions. A system engineer only needs to describe the functionality of the assembly line or characteristics of the product to be assembled, without having to consider further engineering issues. In our application, we successfully used the developed mechanism for planning pick-and-place operations of an industry robot by Kuka as well as the Festo portal robot, when jointly applied for assembling of PCB boards. As future work, we aim to consider product assembly tasks involving more complex products and production layouts.

References

[1] S. Balakirsky and Z. Kootbally. *An Ontology Based Approach to Action Verification for Agile Manufacturing*, pages 201–217. Springer International Publishing, Cham, 2014.

[2] G. A. Bekey. *Autonomous Robots: From Biological Inspiration to Implementation and Control (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[3] L. Buoncompagni, A. Capitanelli, and F. Mastrogiovanni. A ros multi-ontology references services: Owl reasoners and application prototyping issues. *arXiv preprint arXiv:1706.10151*, 2017.

[4] M. Cashmore, M. Fox, D. Long, D. Magazzeni, B. Ridder, A. Carrera, N. Palomeras, N. Hurtos, and M. Carreras. Rosplan: Planning in the robot operating system. In *Twenty-Fifth International Conference on Automated Planning and Scheduling*, 2015.

[5] M. Crosby, R. P. A. Petrick, F. Rovida, and V. Krüger. Integrating mission and task planning in an industrial robotics framework. In *ICAPS*, 2017.

[6] D. Dou. The formal syntax and semantics of webpddl. Technical report, Technical Report, Technical report, University of Oregon, 2008.

[7] T. Hoebert, W. Lepuschitz, E. List, and M. Merdan. Cloud-based digital twin for industrial robotics. pages 105–116, Cham, 2019. Springer International Publishing.

[8] A. Perzylo, N. Somani, S. Profanter, I. Kessler, M. Rickert, and A. Knoll. Intuitive instruction of industrial robots: Semantic process descriptions for small lot production. 10 2016.